



Real-time Prototypes

Ken Fregeolle

24 August 1995

Real-time Prototypes



Prototype Overview

- **Resource Management Subsystem**
 - **String Manager**
- **Command Subsystem**
 - **Telecommand Codeblock Cyclic Redundancy Check**
 - **Frame Operation Procedure Protocol**

Real-time Prototypes



Resource Management Subsystem

- **String Manager Prototype**

- **Purpose**

- Validate Post-PDR distributed RMS concept and design**

- Evaluate inter-task and inter-CPU communication mechanisms**

- IPC via Hughes Class Library/HMessage**

- IPC via Rogue Wave/socket**

- **Approach**

- Create a logical string using separate Server and Workstation tasks**

- Initiate Telemetry Decom prototype via RMS**

- Implement first-phase prototyping using HCL**

- Implement second-phase using Rogue Wave**

Real-time Prototypes



Resource Management Subsystem

- **String Manager Prototype (cont'd.)**

- **Results**

- Refined RMS subsystem task definitions**

- Integrated distributed RMS concept into subsystem critical design**

- Incorporated H/W & S/W status monitoring into a single Resource Monitor**

- Devised RMS “smart request” objects**

- Requests to RMS which execute themselves**

- **Future Goals**

- Prototype Active/Backup string management**

- Prototype Resource Monitor task**

Real-time Prototypes



Command Subsystem

- **AM-1 Cyclic Redundancy Check Prototype**
 - **Purpose**
Evaluate potential CRC mechanisms and implementations
 - **Approach**
Simulate AM-1 ICD-106 model for Telecommand Codeblock encoding
 - **Results**
Developed reusable, generic CRC class
(for production use)
Devised virtual CRC algorithm concept
Assisted spacecraft manufacturer regarding expected CRC results

Real-time Prototypes



Command Subsystem

- **Frame Operation Procedure Prototype**

- **Purpose**

- Demonstrate FOP-1 functionality via software state-machine**
 - Prototype CLCW interpretation and spacecraft/ground synchronization**

- **Approach**

- Evaluate FOP-1 state-table alternatives**

- Array of function pointers**

- Class hierarchy of states**

- Create tool for testing FOP-1 protocol**

- Create CLCWs**

- Generate 26 of 41 potential FOP-1 events**

- Utilize CSS API for inter-task communication**

Real-time Prototypes



Command Subsystem

- **Frame Operation Procedure Prototype (cont'd.)**

- **Results**

- Selected class-based state machine for use in subsystem critical design**

- **Future Goals**

- Build entire command through CLTU based upon command mnemonic**

- Simulate transmission of CLTU**

- Integrate CLTU and FOP prototypes for closed-loop testing**